

苏州大学实验报告

院、系	计算机学院	姓名	朱金涛	学号	2327406014
课程名称	计算机网络				
指导教师	高国举		实验完成日期	2025年12月2日	

实验名称：网际协议

一、实验目的

- 掌握 IP 数据报的报文格式
- 掌握 IP 校验和计算方法
- 掌握子网掩码和路由转发
- 理解特殊 IP 地址的含义
- 理解 IP 分片过程
- 理解协议栈对 IP 协议的处理方法
- 理解 IP 路由表作用以及 IP 路由表的管理

二、实验步骤与结果

练习 1：编辑并发送 IP 数据报

各主机打开协议分析器，进入相应的网络结构并验证网络拓扑的正确性，如果通过拓扑验证，关闭协议分析器继续进行实验，如果没有通过拓扑验证，请检查网络连接。本练习将主机 A、B、C、D、E、F 作为一组进行实验。

1. 主机 B 在命令行方式下输入 staticroute_config 命令，开启静态路由服务。

```
C:\> 管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.17763.107]
(c) 2018 Microsoft Corporation。保留所有权利。
C:\> staticroute_config
[SC] ChangeServiceConfig 成功

SERVICE_NAME: remoteaccess
  TYPE               : 20  WIN32_SHARE_PROCESS
  STATE              : 2   START_PENDING
                      (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
  WIN32_EXIT_CODE    : 0   (0x0)
  SERVICE_EXIT_CODE : 0   (0x0)
  CHECKPOINT        : 0x0
  WAIT_HINT         : 0x7d0
  PID                : 5328
  FLAGS
```

2. 主机 A 启动协议编辑器，编辑一个 IP 数据报，其中：

MAC 层：

目的 MAC 地址：主机 B 的 MAC 地址（对应于 172.16.1.1 接口的 MAC）。

源 MAC 地址：主机 A 的 MAC 地址。

协议类型或数据长度：0800。

IP 层：

总长度：IP 层长度。

生存时间：128。

源 IP 地址：主机 A 的 IP 地址（172.16.1.2）。

目的 IP 地址：主机 E 的 IP 地址（172.16.0.2）。

校验和：在其它所有字段填充完毕后计算并填充。

自定义字段：

数据：填入大于 1 字节的用户数据。

【说明】先使用协议编辑器的“手动计算”校验和，再使用协议编辑器的“自动计算”校验和，将两次计算结果相比较，若结果不一致，则重新计算。

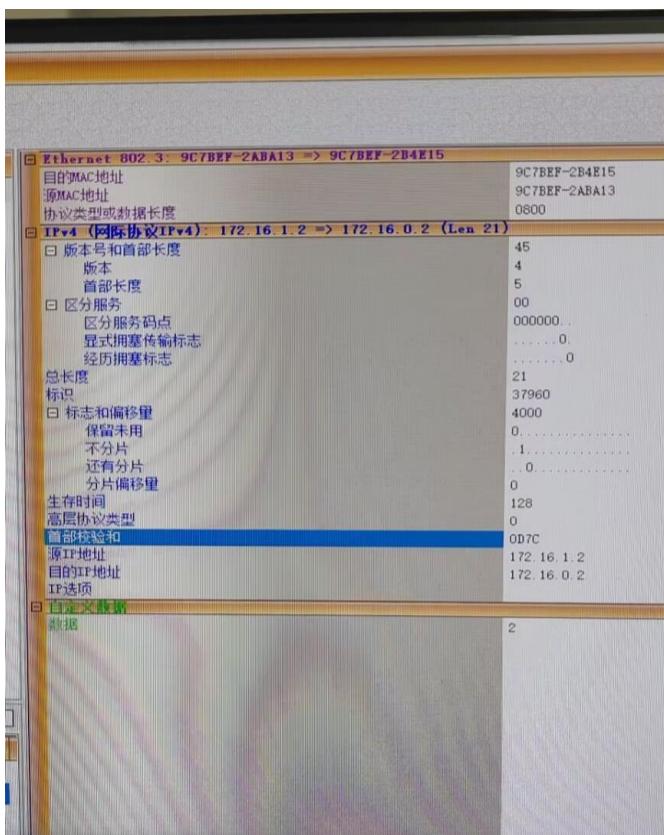
- IP 在计算校验和时包括哪些内容？

答案：IP 校验和只校验 IP 数据报的首部不包括数据部分。具体包括：版本、首部长度、服务类型、总长度、标识、标志、片偏移、生存时间（TTL）、协议、源 IP 地址、目

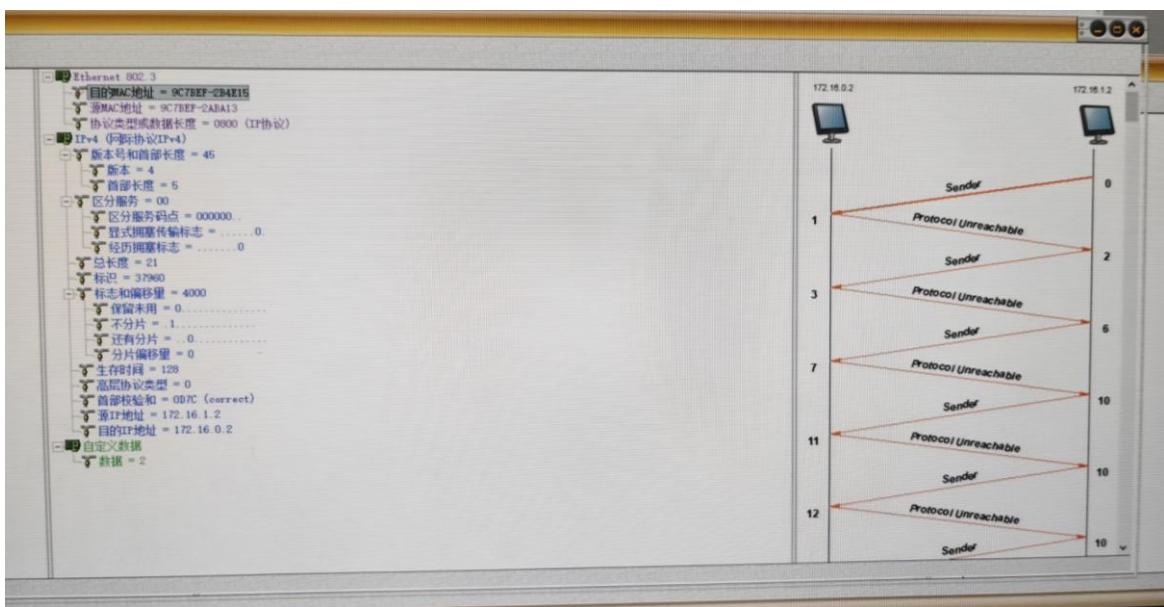
的 IP 地址。

3. 在主机 B (两块网卡分别打开两个捕获窗口)、E 上启动协议分析器，设置过滤条件 (提取 IP 协议)，开始捕获数据。
4. 主机 A 发送第 1 步中编辑好的报文。
5. 主机 B、E 停止捕获数据，在捕获到的数据中查找主机 A 所发送的数据报，并回答以下问题：

主机 A 编辑的报文：



主机 B 接收到的信息：



主机 E 接收到的信息：

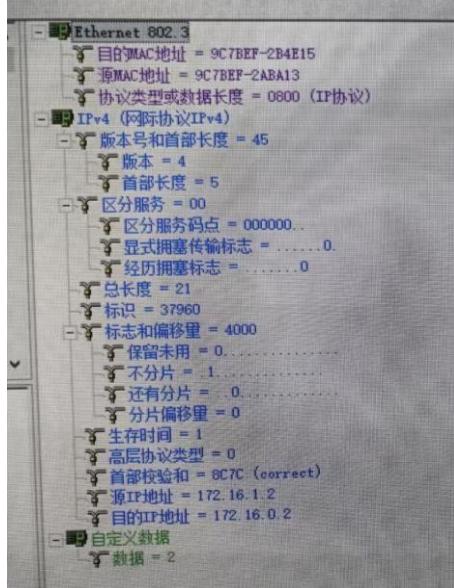
第1步中主机A所编辑的报文，经过主机B到达主机E后，报文数据是否发生变化？

若发生变化，记录变化的字段，并简述发生变化的原因。

答案：发生变化。

1. 生存时间 (TTL): 减 1。原因：路由器每转发一次数据报，TTL 值减 1，防止数据报在网络中无限循环。
2. 首部校验：发生改变。原因：由于 TTL 字段发生了变化，路由器必须重新计算 IP 首部校验和。
3. 源 MAC 地址和目的 MAC 地址：发生改变。原因：数据链路层帧头在每一跳都会重写，源 MAC 变为 B 的 MAC，目的 MAC 变为 E 的 MAC。
6. 将第1步中主机A所编辑的报文的“生存时间”设置为1，重新计算校验和。
7. 主机B、E重新开始捕获数据。
8. 主机A发送第5步中编辑好的报文。
9. 主机B、E停止捕获数据，在捕获到的数据中查找主机A所发送的数据报，并回答以下问题：

主机B第二次捕获到的数据信息：



序号	源地址	目的地址	概要	帧长度	时间	时间间隔
□□□0	9C7BEF-2BAE13	9C7BEF-2BAE15	IPv4 (网际协议IPv4): 172.16.1.2 => 1... 60	60	16:22:04.01...	
□□□1	172.16.1.2	172.16.0.2	ICMP (Internet控制报文协议) 数据包超时 63	63	16:22:04.01...	0.0000000
□□□2	9C7BEF-2BAE15	9C7BEF-2BAE13	IPv4 (网际协议IPv4): 172.16.1.2 => 1... 60	60	16:22:05.02...	1.0000000
□□□3	172.16.1.1	172.16.1.2	ICMP (Internet控制报文协议) 数据包超时 63	63	16:22:05.02...	0.0000000
□□□4	9C7BEF-2BAE13	9C7BEF-2BAE15	IPv4 (网际协议IPv4): 172.16.1.2 => 1... 60	60	16:22:06.02...	1.0000000
□□□5	172.16.1.1	172.16.0.2	ICMP (Internet控制报文协议) 数据包超时 63	63	16:22:06.02...	0.0000000
□□□6	9C7BEF-2BAE15	9C7BEF-2BAE13	IPv4 (网际协议IPv4): 172.16.1.2 => 1... 60	60	16:22:07.03...	1.0000000
□□□7	172.16.1.1	172.16.1.2	ICMP (Internet控制报文协议) 数据包超时 63	63	16:22:07.03...	0.0000000

00000000: 9C 7B EF 2B 4E 15 9C 7B EF 2A BA 13 08 00 45 00 .(i+N.{i^o...E.
00000010: 00 15 94 48 40 00 01 00 8C 7C AC 10 01 02 AC 10 ...H@...|-.-.-.
00000020: 00 02 32 00 00 00 00 00 00 00 00 00 00 00 00 00 ..2
00000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

主机 E 第二次接收到的信息：

序号	源地址	目的地址	数据	帧长	时间	时间间隔
□□□46	DC534B-0E8373 333300-00004	透明连接桥头 <- 10.0.0.14	90	16:21:21.08...0.0000000		
□□□47	FE00:0000:0000:0000:0000:0000:0000:0000	ICMPv4协议 - 未知的ICMPv4类型	90	16:21:21.08...0.0000000		
□□□48	DC534B-0E8373 01000E-00007B	ICMPv4(用户数据报协议): Port 5353 => 5353 71	16:21:21.08...0.0000000			
□□□49	172.16.0.1 224.0.6.251	ICMPv4(用户数据报协议): Port 5353 => 5353 91	16:21:21.08...0.0000000			
□□□50	DC534B-0E8373 333300-00007B	ICMPv4(用户数据报协议): Port 5353 => 5353 109	16:21:21.08...0.0000000			
□□□51	172.16.0.2 224.0.6.251	ICMPv4(用户数据报协议): Port 5353 => 5355 85	16:21:21.08...0.0000000			
□□□52	DC534B-0E8373 333300-000083	ICMPv4(用户数据报协议): Port 56047 => 5355 45	16:21:21.08...0.0000000			
□□□53	172.16.0.1 224.0.6.252	ICMPv4(用户数据报协议): Port 56047 => 5355 116	16:21:21.08...0.0000000			

00000000: 01 00 SE 00 00 FB 2C 5A 08 E9 73 09 00 45 00 .~.0.S2.8e.
00000010: 00 39 85 C9 00 00 01 11 A6 DE AC 10 00 01 E0 00 .9.E...|D...-.
00000020: 00 F8 18 E9 14 E9 00 25 E9 D1 00 00 00 00 01 .E.e.e.e8...
00000030: 00 00 00 00 00 00 05 48 50 34 33 39 05 EC 6F 63 .HF439.loc
00000040: 61 6C 00 FF 09 01 al..y..

Ethernet 802.3

- 源MAC地址 = 0000E0-0000F3
- 目的MAC地址 = 20534A-0E8373
- 协议类型或数据长度 = 0800 (IP协议)
- IP4 (网际协议IPv4)
- 源本机和目的长度 = 46
 - 源长度 = 4
 - 目的长度 = 5
- 分片偏移 = 0
- 分片数或报文ID = 00000
- 翻转或塞件头标志 = 0
- 经历或塞件标志 = 0
- 总长度 = 57
- 标识 = 34249
- 标志和源端口号 = 0000
- 保留未用 = 0
- 不分片 = 0
- 碎片偏移 = 0
- 分片数标志 = 0
- 生存时间 = 1
- 基础协议类型 = 17 (ICMP协议)
- 首部校验和 = A60E (correct)
- 源IP地址 = 172.16.0.1
- 目的IP地址 = 224.0.0.251
- ICMP (用户数据报协议)
 - 源端口 = 5633
 - 目的端口 = 5353
 - 协议数据长度 = 37
 - 检查和 = E901 (correct)
- 选定的数据
 - 数据 =

RP399 local

- 主机 B、E 是否能捕获到主机 A 所发送的报文？简述产生这种现象的原因。

答案：

- **主机 B:** 能捕获到。
- **主机 E:** 不能捕获到。
- **原因:** 主机 A 发送的报文 TTL 为 1。当报文到达路由器（主机 B）时，B 将 TTL 减 1 变为 0。根据 IP 协议规定，当 TTL 为 0 时，路由器会丢弃该报文，并向源主机发送 ICMP“超时”差错报文，因此报文不会被转发给主机 E。

练习 2：特殊的 IP 地址

本练习将主机 A、B、C、D、E、F 作为一组进行实验。

1. 直接广播地址

(1) 主机 A 编辑 IP 数据报 1, 其中:

目的 MAC 地址: FFFFFF-FFFFFF。

源 MAC 地址: A 的 MAC 地址。

源 IP 地址: A 的 IP 地址。

目的 IP 地址: 172.16.1.255。

自定义字段数据: 填入大于 1 字节的用户数据。

校验和: 在其它字段填充完毕后, 计算并填充。

(2) 主机 A 再编辑 IP 数据报 2, 其中:

目的 MAC 地址: 主机 B 的 MAC 地址 (对应于 172.16.1.1 接口的 MAC)。

源 MAC 地址: A 的 MAC 地址。

源 IP 地址: A 的 IP 地址。

目的 IP 地址: 172.16.0.255。

自定义字段数据: 填入大于 1 字节的用户数据。

校验和: 在其它字段填充完毕后, 计算并填充。

(3) 主机 B、C、D、E、F 启动协议分析器并设置过滤条件 (提取 IP 协议, 捕获

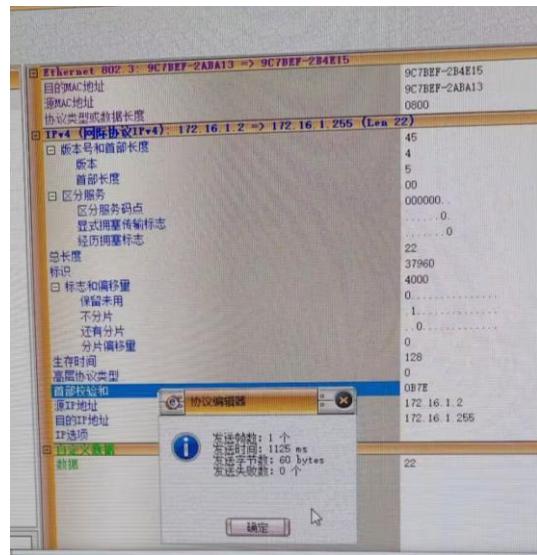
172.16.1.2 接收和发送的所有 IP 数据包, 设置地址过滤条件如下: 172.16.1.2<->Any)。

(4) 主机 B、C、D、E、F 开始捕获数据。

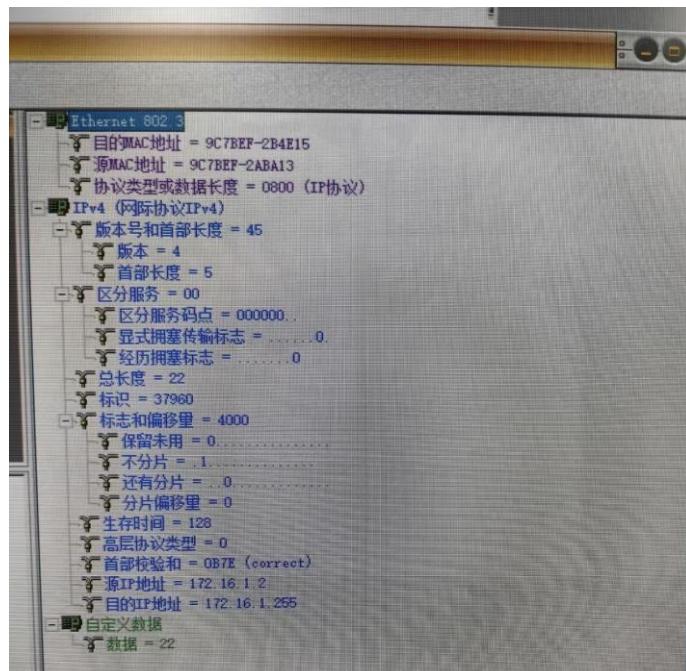
(5) 主机 A 同时发送这两个数据报。

(6) 主机 B、C、D、E、F 停止捕获数据。

主机 A 编辑发送的 IP 数据报 1:

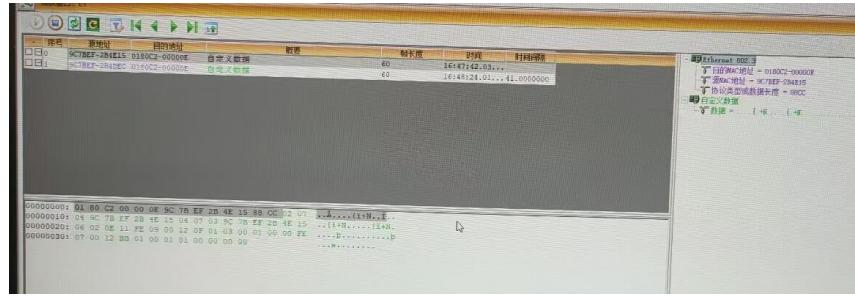


主机 B:

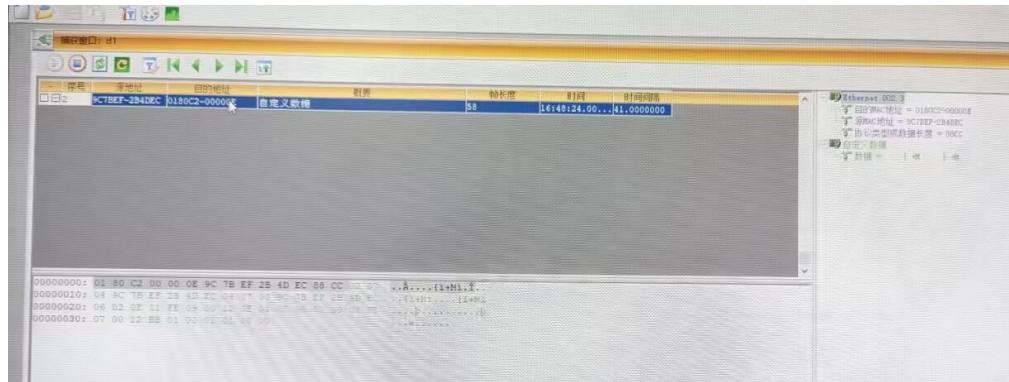


序号	源地址	目的地址	概要	帧长度	时间	时间间隔
0	9C7BEF-2ABA13 172.16.1.2	9C7BEF-2B4E15 172.16.1.255	IPv4 (网际协议IPv4): 172.16.1.2 => 1...	60	16:49:39.07...	

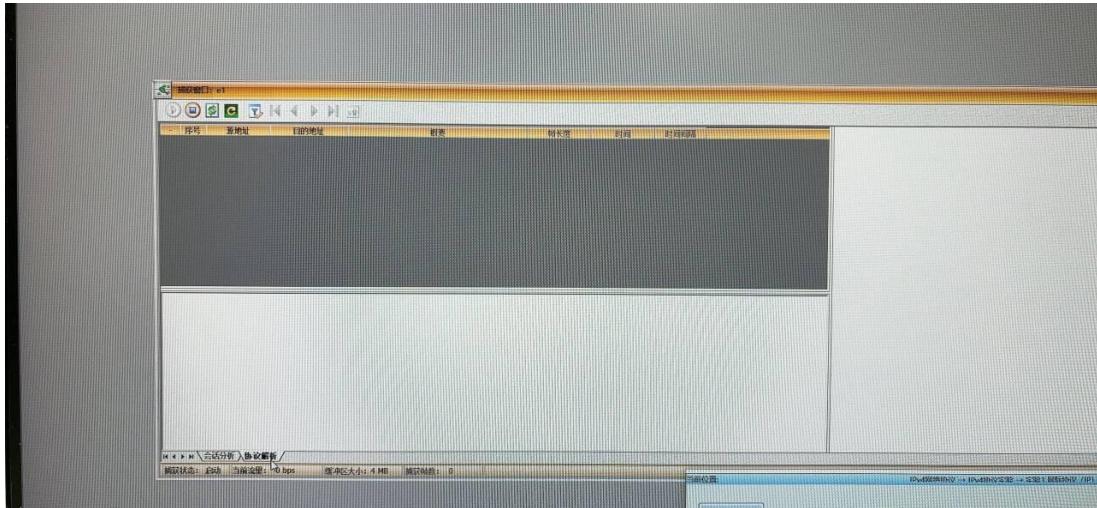
主机 C:



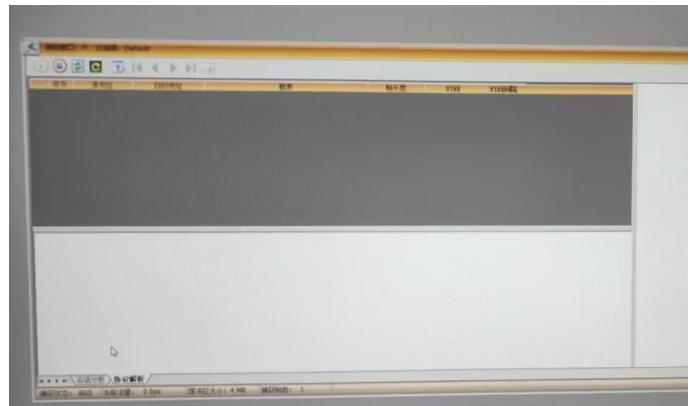
主机 D:



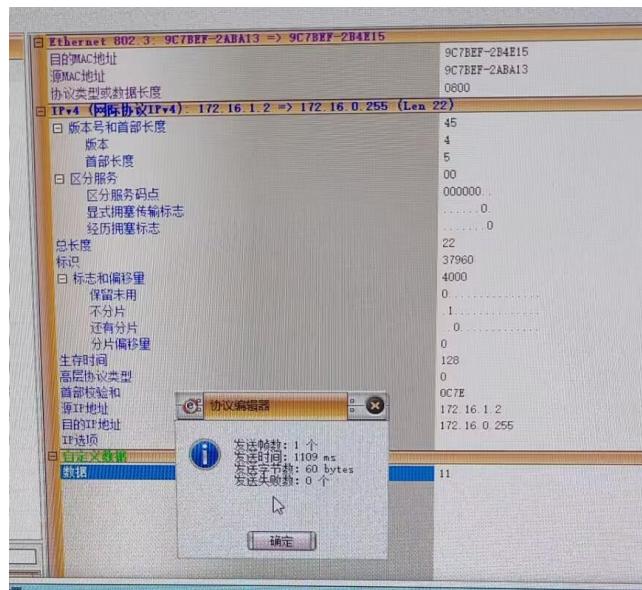
主机 E:



主机 F:



主机 A 编辑发送的数据报 2:



主机 B:

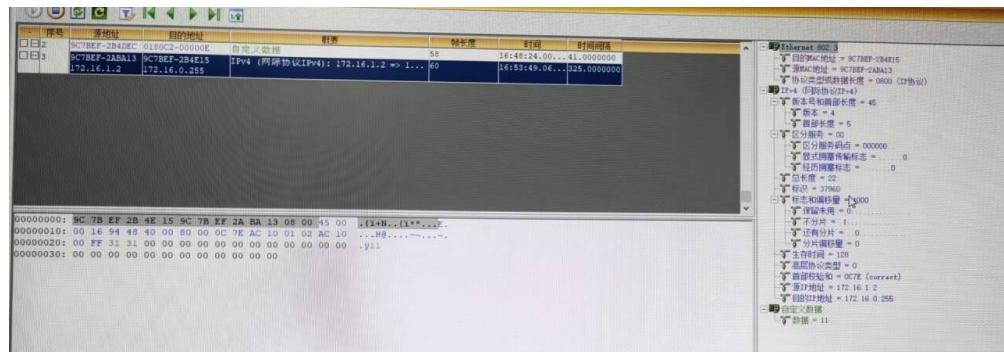
文件窗口: 未命名5

序号	源地址	目的地址	摘要	帧长度	时间	时间间隔
0	9C7B EF-2B A1 3	9C7B EF-2B 4E 15	IPv4 (网际协议IPv4): 172.16.1.2 => 1...	60	16:53:59.05...	
	172.16.1.2	172.16.0.255				

0000000: 9C 7B EF 2B 4E 15 9C 7B EF 2A BA 13 08 00 45 00 .{i+N..{i^o...E
0000010: 00 16 94 48 40 00 80 00 0C 7E AC 10 01 02 AC 10 ..H8....-+--
0000020: 00 FF 31 31 00 00 00 00 00 00 00 00 00 00 00 00 00 ..y...
0000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..

主机 E:

主机 F:



C、D 并未收到。

● 记录实验结果

	主机号
收到 IP 数据报 1	B、C、D
收到 IP 数据报 2	B、E、F

(注：IP 数据报 1 是发给 A 所在子网的广播，同网段的 B、C、D 都能收到。IP 数据报 2 是发给另一子网的广播，主机 A 将其发给网关 B，B 会收到，由于开启了定向广播转发功能，所以 E、F 也能收到。)

● 结合实验结果，简述直接广播地址的作用。

答案： 直接广播地址（HostID 全为 1）用于向指定的某个网络上的所有主机发送数据报。

它允许远程主机向另一个网络的所有主机广播消息。

2. 受限广播地址

(1) 主机 A 编辑一个 IP 数据报，其中：

目的 MAC 地址：FFFFF-FFFFF。

源 MAC 地址：A 的 MAC 地址。

源 IP 地址：A 的 IP 地址。

目的 IP 地址：255.255.255.255。

自定义字段数据：填入大于 1 字节的用户数据。

校验和：在其它字段填充完毕后，计算并填充。

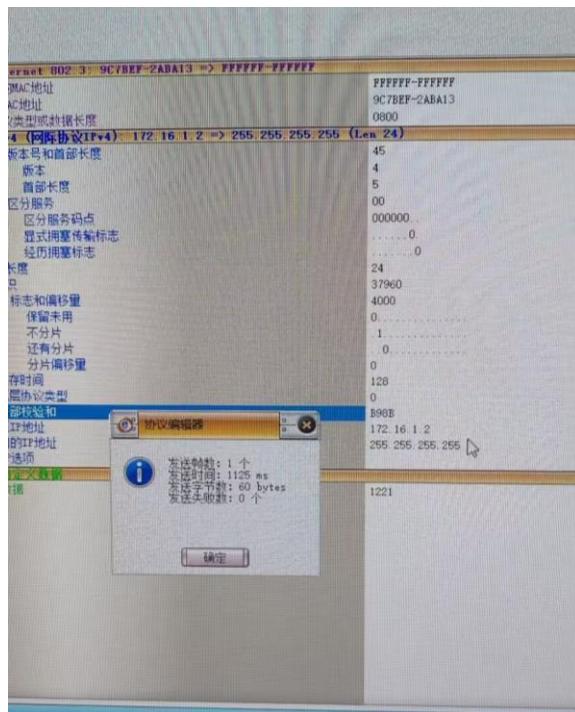
(2) 主机 B、C、D、E、F 重新启动协议分析器并设置过滤条件（提取 IP 协议，捕获 172.16.1.2 接收和发送的所有 IP 数据包，设置地址过滤条件如下：172.16.1.2<->Any）。

(3) 主机 B、C、D、E、F 重新开始捕获数据。

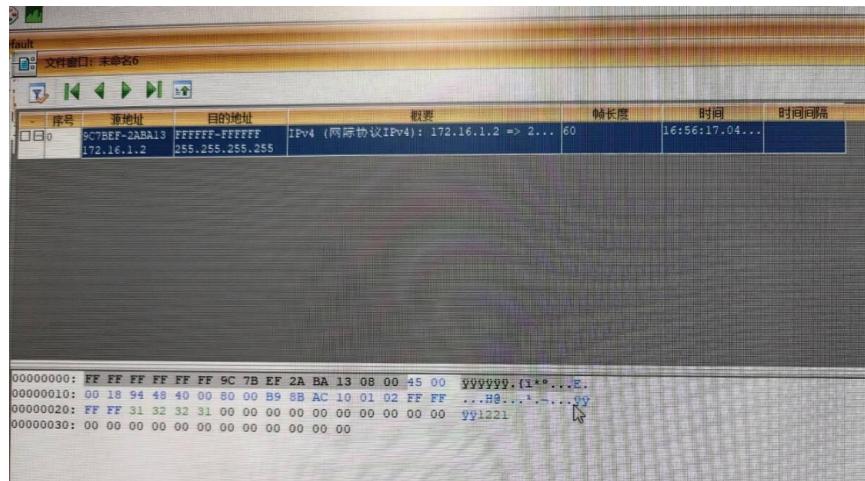
(4) 主机 A 发送这个数据报。

(5) 主机 B、C、D、E、F 停止捕获数据。

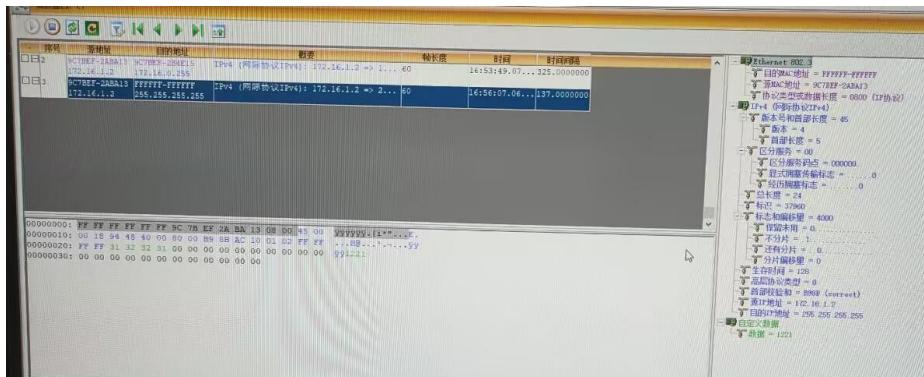
A 发送的数据报：



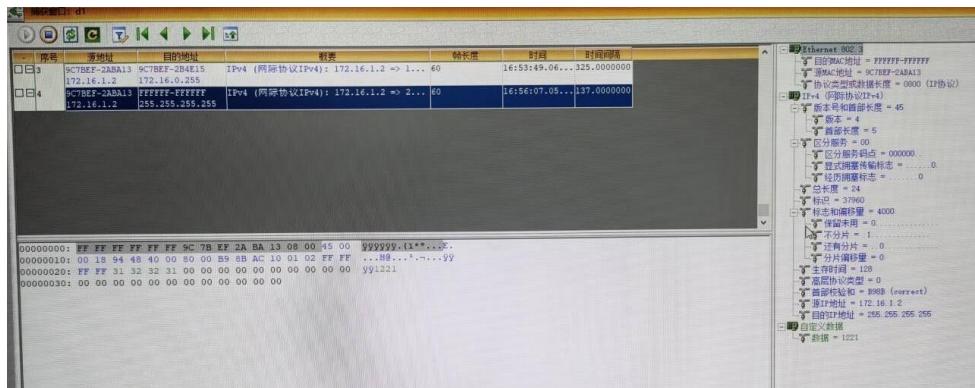
主机 B：



主机 C:



主机 D:



E、F 并未收到。

● 记录实验结果

	主机号
收到主机 A 发送的 IP 数据 报	B、C、D
未收到主机 A 发送的 IP 数 据报	E、F

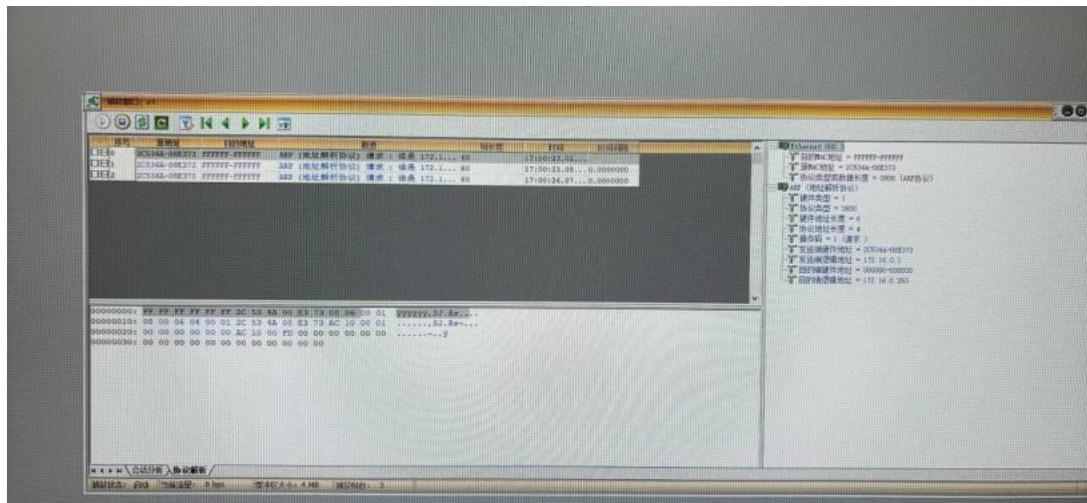
● 结合实验结果，简述受限广播地址的作用。

答案： 受限广播地址（255.255.255.255）用于向本物理网络（也就是发送方所在的本地局域网）上的所有主机广播数据。路由器不会转发目的地址为受限广播地址的数据报，因此广播被限制在本地链路范围内。

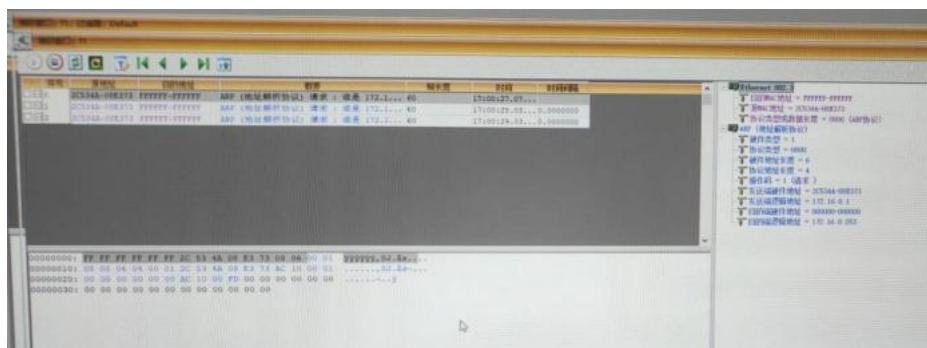
3. 环回地址

- (1) 主机 F 重新启动协议分析器开始捕获数据并设置过滤条件 (提取 IP 协议)。
- (2) 主机 E ping 127.0.0.1。
- (3) 主机 F 停止捕获数据。

主机 E:



主机 F:



- 主机 F 是否收到主机 E 发送的目的地址为 127.0.0.1 的 IP 数据报？为什么？

答案：主机 F 不会收到目的地址为 127.0.0.1 的 IP 数据报。因为 127.0.0.1 属于环回地址，报文不会真正发到网卡上，而是在主机 E 内部的协议栈中完成发送与接收，相当于“自己跟自己通信”。

练习 3：IP 数据报分片

本练习将主机 A、B、C、D、E、F 作为一组进行实验。

1. 在主机 B 上使用“实验平台上工具栏中的 MTU 工具”设置以太网端口的 MTU 为 800 字节（两个端口都设置）。

The screenshot shows a Windows Command Prompt window titled '管理员: C:\Windows\SYSTEM32\cmd.exe'. The user runs 'netsh interface ipv4 show subinterfaces' to view current MTU settings. It lists several interfaces with their MTUs set to 1500. Then, the user runs two commands to change the MTU of subinterfaces 'b1' and 'b2' to 800, using 'store=persistent' to save the changes. Finally, they run 'netsh interface ipv4 show subinterfaces' again to verify the changes.

```
c:\>netsh interface ipv4 show subinterfaces
MTU MediaSenseState 传入字节 传出字节 接口
4294967295 1 856979 0 9870 Loopback Pseudo-Interface 1
orkstat: 1500 1 0 23341 b1
1500 1 0 15487 VMware Network Adapter VMnet1
1500 1 0 15407 VMware Network Adapter VMnet8
1500 1 9846 16509 b2

c:\>netsh interface ipv4 set subinterface "b1" mtu=800 store=persistent
确定。

c:\>netsh interface ipv4 set subinterface "b2" mtu=800 store=persistent
确定。

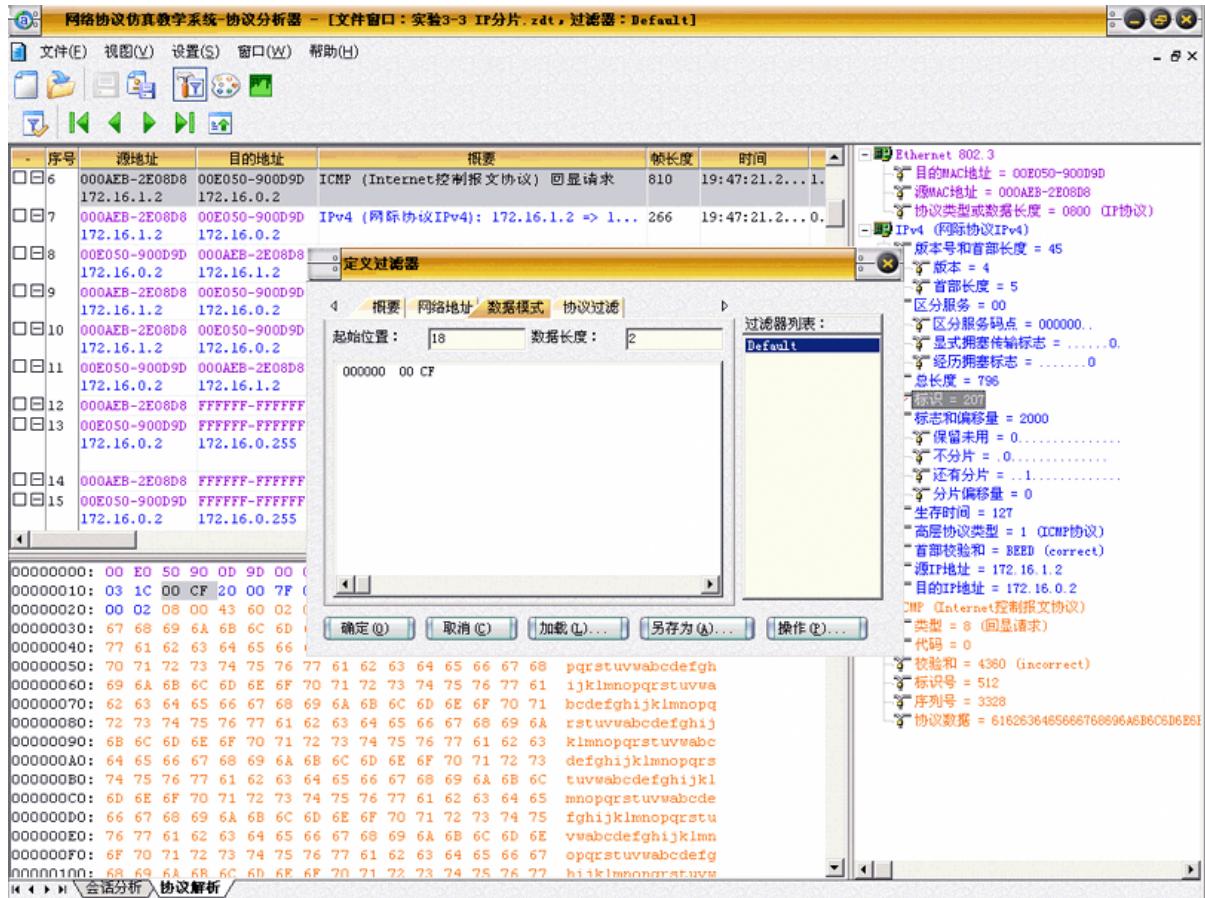
c:\>netsh interface ipv4 show subinterfaces
MTU MediaSenseState 传入字节 传出字节 接口
4294967295 800 1 856979 0 9870 Loopback Pseudo-Interface 1
800 1 0 23341 b1
1500 1 0 15487 VMware Network Adapter VMnet1
1500 1 0 15407 VMware Network Adapter VMnet8
800 1 9846 16509 b2
```

2. 主机 A、B、E 启动协议分析器，打开捕获窗口进行数据捕获并设置过滤条件(提取 ICMP 协议)。
3. 在主机 A 上，执行命令 ping -1 1000 172.16.0.2。

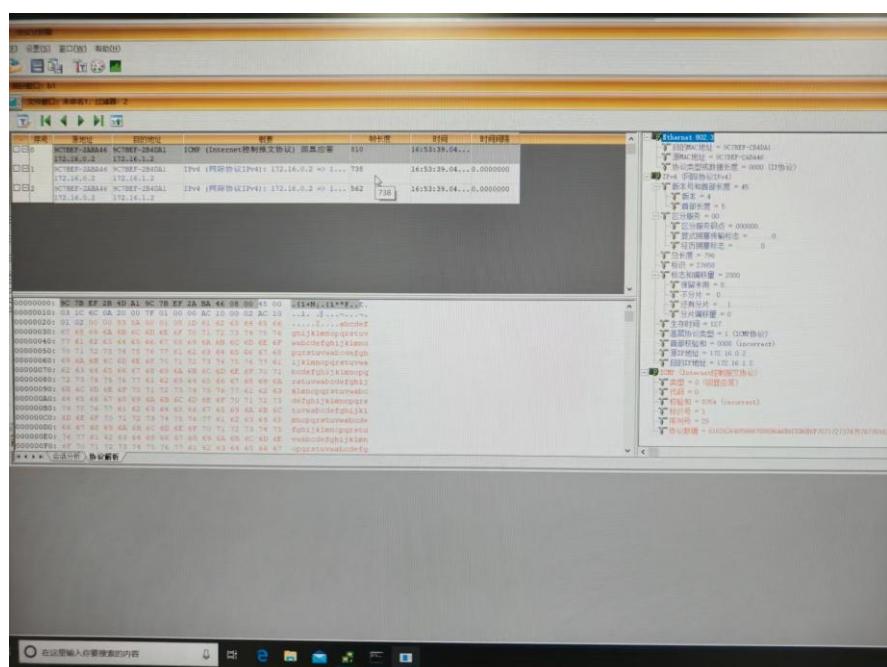
The screenshot shows a Windows Command Prompt window titled '管理员: C:\Windows\SYSTEM32\cmd.exe'. The user runs 'ping -1 1000 172.16.0.2' to perform a ping test. The output shows four ping requests being sent to the target IP address. Each request has a size of 1000 bytes. The statistics for each ping show 4 transmitted, 0 received, and 4 lost (100% loss). The process is repeated three times.

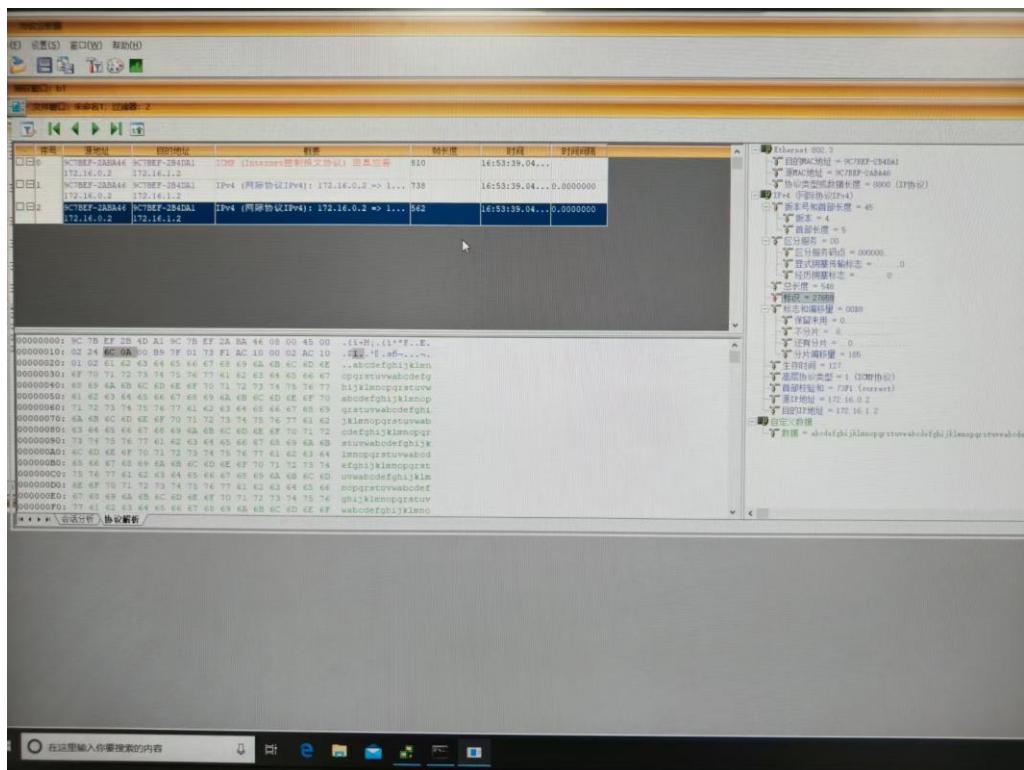
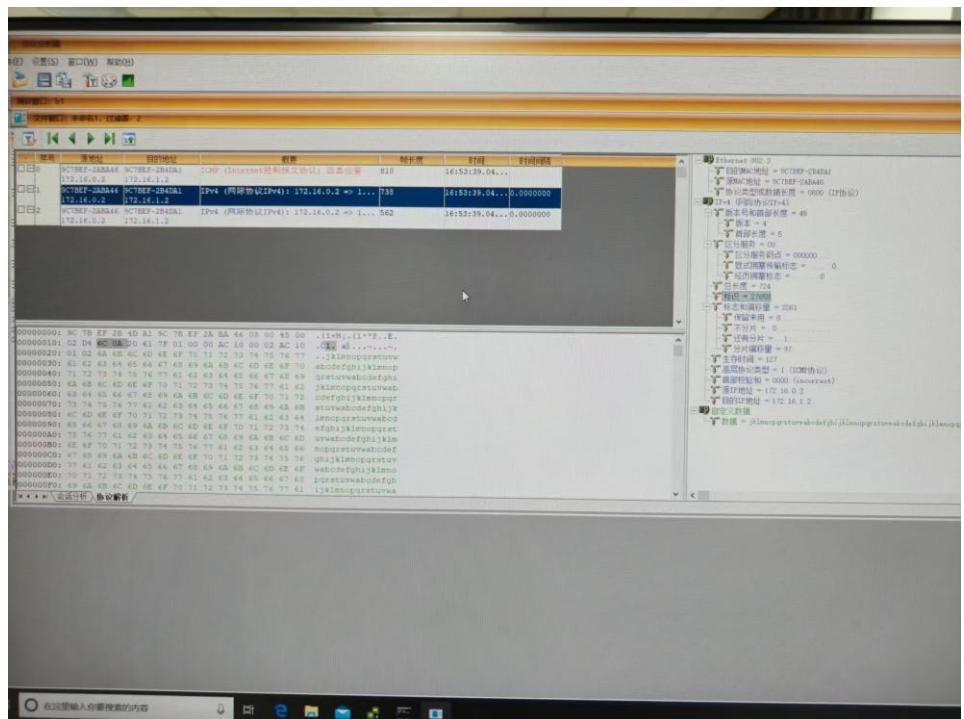
```
c:\>ping -1 1000 172.16.0.2
正在 Ping 172.16.0.2 具有 1000 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。
172.16.0.2 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 0, 丢失 = 4 (100% 丢失).
c:\>ping -1 1000 172.16.0.2
正在 Ping 172.16.0.2 具有 1000 字节的数据:
请求超时。
请求超时。
2 请求超时。
0 请求超时。
172.16.0.2 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 0, 丢失 = 4 (100% 丢失).
c:\>
```

4. 主机 A、B、E 停止捕获数据。在主机 E 上重新定义过滤条件（取一个 ICMP 数据包，按照其 IP 层的标识字段设置过滤），如图所示：



主机 E:



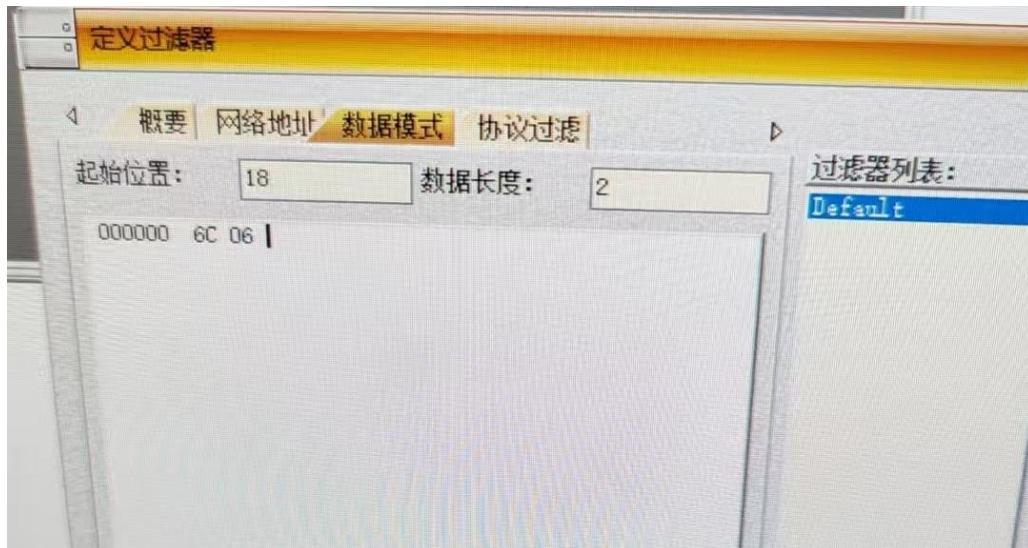


- 将 ICMP 报文分片信息填入下表，分析表格内容，理解分片的过程。

字段名称	分片序号 1	分片序号 2	分片序号 3
“标识”字段值	27658	27658	27658
“还有分片”字段值	1	1	0

“分片偏移量”字段值	0	97	185
传输的数据量	776	704	528

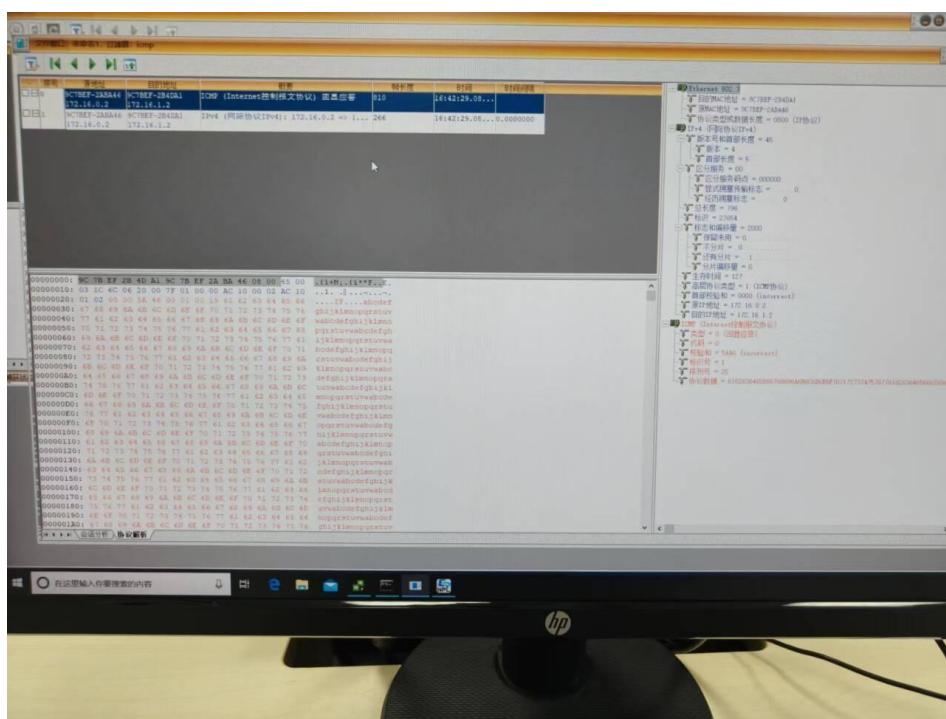
5. 主机 E 恢复默认过滤器。主机 A、B、E 重新开始捕获数据。

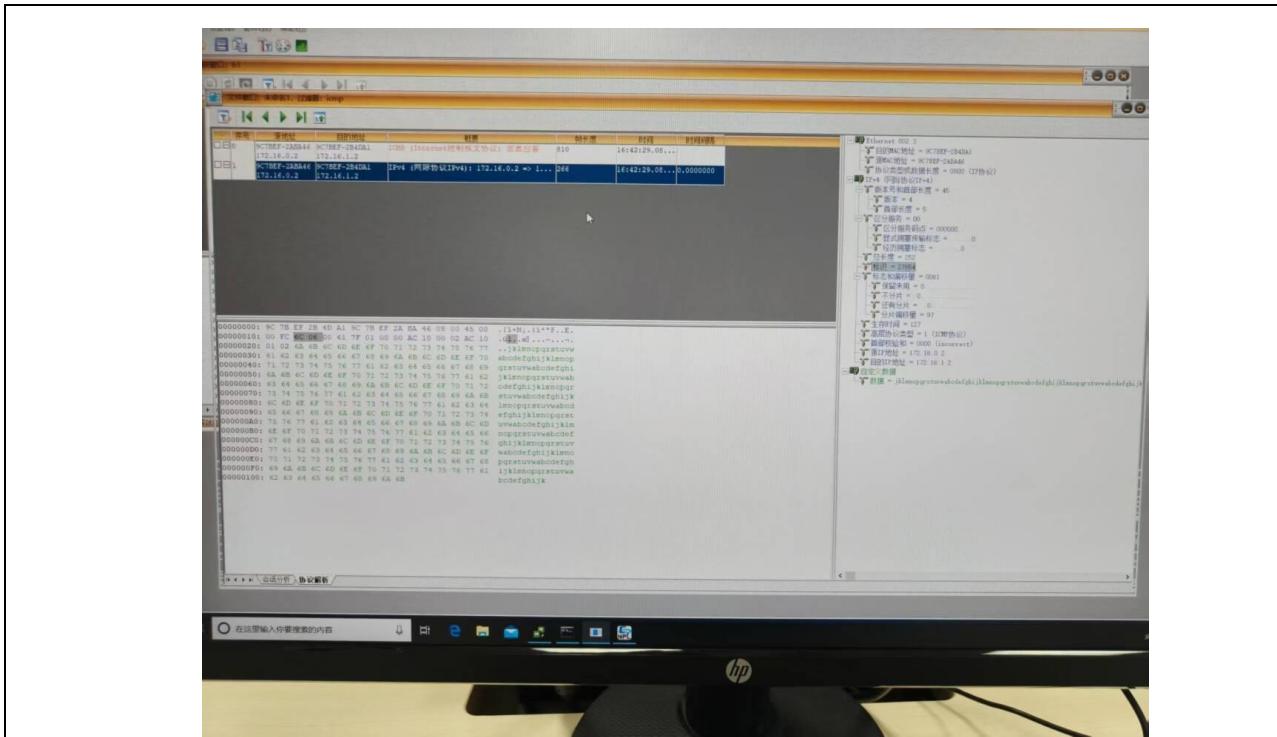


用此过滤后只会捕捉到标志号为 27654 的信息。

6. 在主机 A 上，执行命令 ping -l 2000 172.16.0.2。

主机 E:





7. 主机 A、B、E 停止捕获数据。查看主机 A、E 捕获到的数据，比较两者的差异，体会两次分片过程。

主机 A 与主机 E 捕获数据的差异在于分片的数量和大小，体现了路由传输中的“二次分片”过程。

8. 主机 B 上使用“实验平台上工具栏中的 MTU 工具”恢复以太网端口的 MTU 为 1500 字节。

```
c:\>netsh interface ipv4 set subinterface "b1" mtu=1500 store=persistent
确定。

c:\>netsh interface ipv4 set subinterface "b2" mtu=1500 store=persistent
确定。

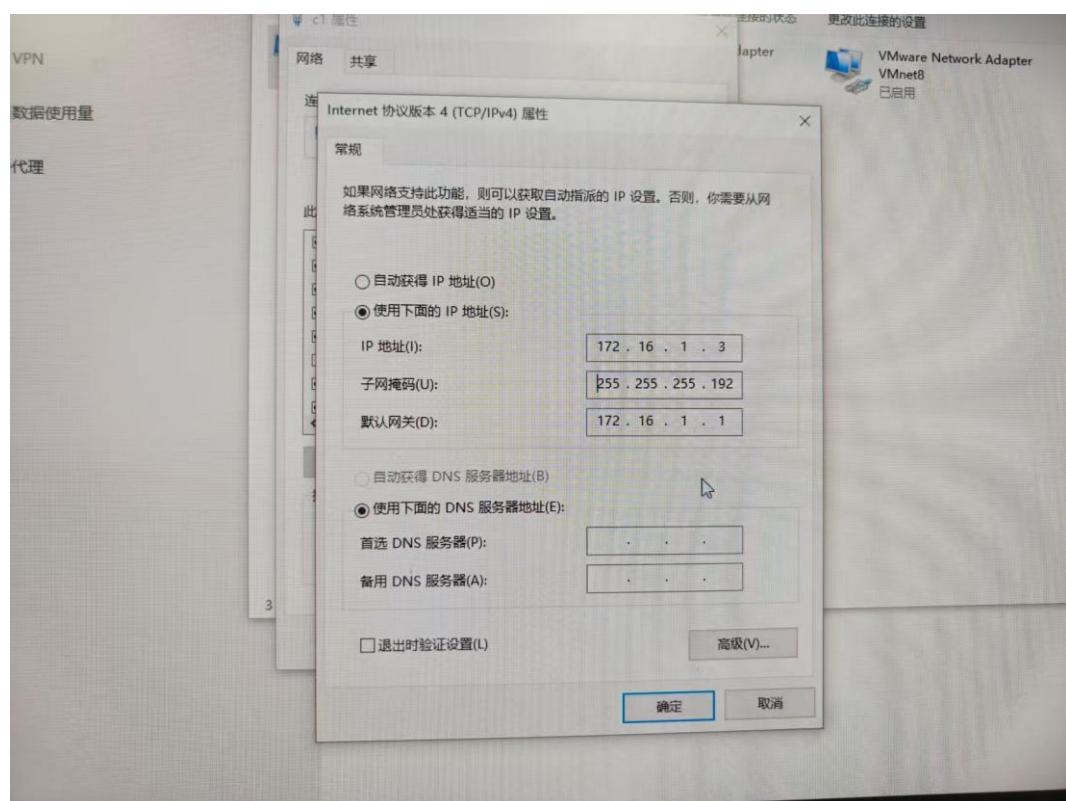
c:\>netsh interface ipv4 show subinterfaces
      MTU  MediaSenseState   传入字节  传出字节      接口
-----  -----  -----
4294967295
    1500          1        911846     39212  b1
    1500          1            0     16632  VMware Network Adapter VMnet1
    1500          1            0     16552  VMware Network Adapter VMnet8
    1500          1        39110     31490  b2
```

练习 4：子网掩码的作用

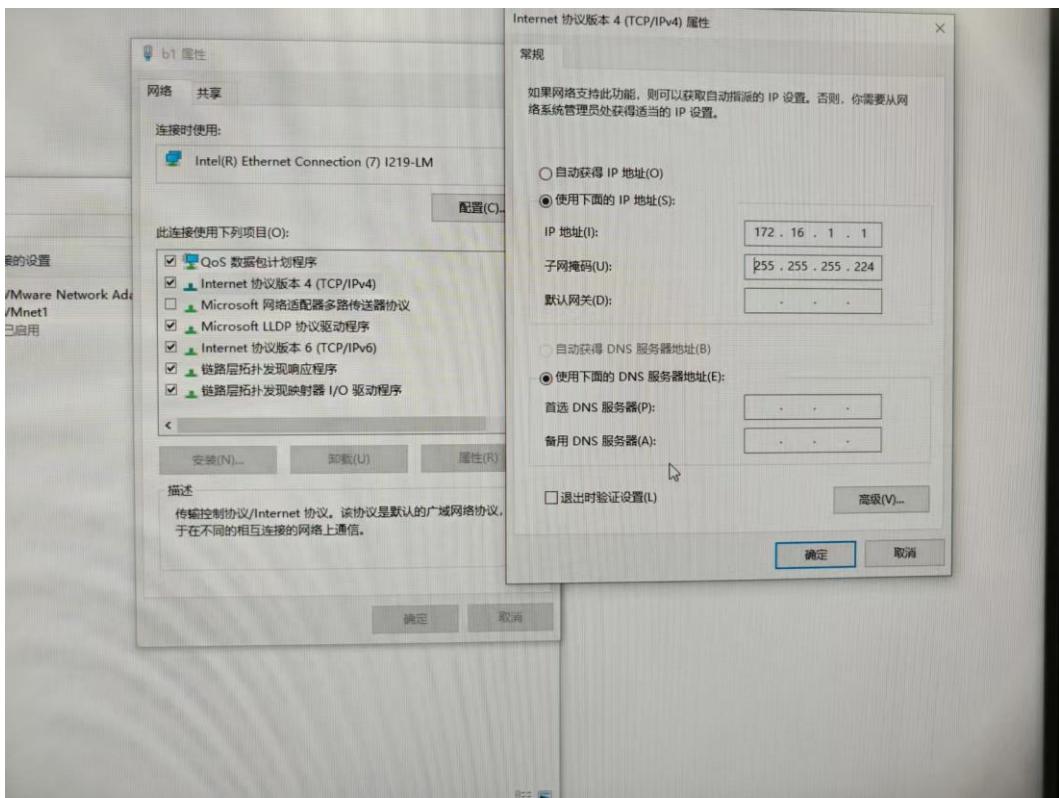
本练习将主机 A、B、C、D、E、F 作为一组进行实验。

1. 所有主机取消网关。
2. 主机 A、C、E 设置子网掩码为 255.255.255.192，主机 B（172.16.1.1）、D、F 设置子网掩码为 255.255.255.224。

主机 A、C、E：

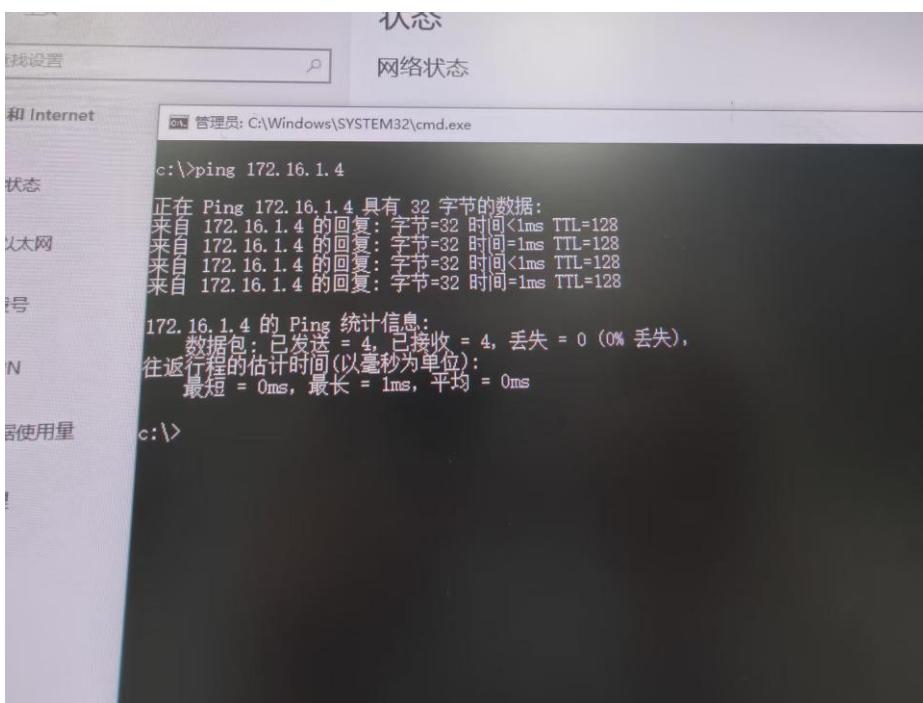


主机 D、F：

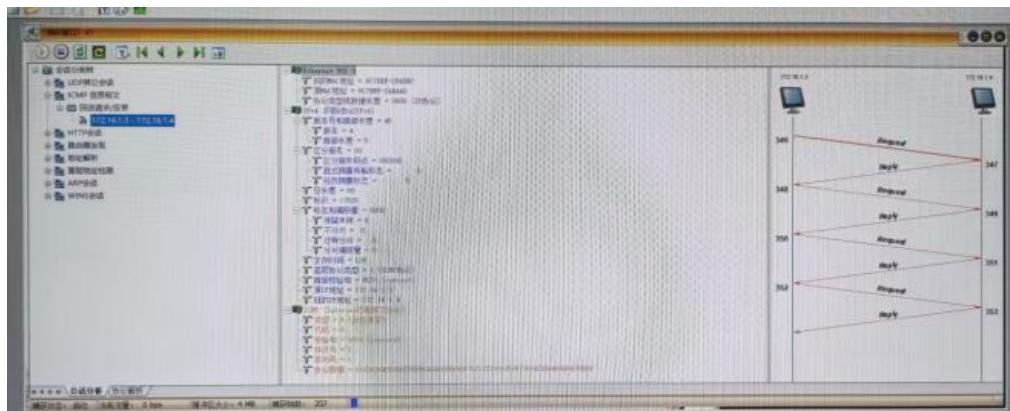


3. 主机 A ping 主机 B (172.16.1.1), 主机 C ping 主机 D (172.16.1.4), 主机 E ping 主机 F (172.16.0.3)。

C ping D:



主机 D:



结果显示可以 Ping 通。

后续 A ping B, E ping F 都能 ping 通。

● 记录实验结果

	是否 ping 通
主机 A----主机 B	可以
主机 C----主机 D	可以
主机 E----主机 F	可以

● 请问什么情况下两主机的子网掩码不同，却可以相互通信？

答：主要取决于它们之间的连接方式与逻辑判断：如果两主机跨越了网段（不在同一广播域），可以通过路由器（网关）进行正常的三层路由转发实现互通；如果两主机在同一物理网络（如同连一台交换机），只要双方的 IP 地址恰好都落在对方子网掩码计算出的“本地地址范围”内（即逻辑上互判为直连），或者网关开启了代理 ARP 功能，它们也能通过直接 ARP 或网关“欺骗”的方式实现通信。

4. 主机 B 在命令行方式下输入 recover_config 命令，停止静态路由服务。
5. 所有主机恢复到网络结构二的配置。

三、实验总结与收获

本次实验通过对 IP 数据报的编辑、发送、捕获及详细分析，使我从理论层面走向实践，深入理解了 IP 协议在网络层的工作机制。具体收获如下：

1. 深入理解 IP 数据报的转发机制与报头变化 通过练习 1，我直观地观察到了数据报在经过路由器（主机 B）转发时，**TTL（生存时间）字段会自动减 1**，且由于 TTL 的变化，**首部校验和会被路由器重新计算**。同时，我也验证了在数据链路层，**源 MAC 和目的 MAC 地址在每一跳都会被重写**，而 IP 头部中的源 IP 和目的 IP 保持不变。这让我深刻体会到了网络层“端到端”传输与数据链路层“逐跳”传输的区别。
2. 明确了特殊 IP 地址的应用场景与传输范围 在练习 2 中，通过对比实验，我清晰地区分了**受限广播地址（255.255.255.255）**和**直接广播地址**的区别：前者被限制在本地物理网络内，路由器不予转发；后者则可以跨越路由器传播。此外，对**环回地址**（127.0.0.1）的测试让我明白这类报文仅在本地协议栈内部循环，不会经过物理网卡，这对于理解网络测试与本地通信至关重要。
3. 掌握了 MTU 与 IP 分片的具体过程 练习 3 是本次实验的难点也是亮点。通过将路由器 MTU 设置为 800 字节并发送大包，我捕获并分析了分片报文。我观察到了“**二次分片**”现象（主机 A 先分片，路由器主机 B 因 MTU 更小再次分片），并学会了如何通过 IP 报头中的“**标识**”、“**标志（MF 位）**”和“**片偏移**”字段来追踪和重组原始数据。这使我对网络层如何适应不同链路 MTU 有了具象的认识。
4. 辩证理解子网掩码在通信中的作用 练习 4 打破了我对子网掩码的固有认知。实验证明，即使两台主机的子网掩码不同，只要双方 IP 地址在各自的逻辑运算中都被判断为“**直连范围**”内，或者通过网关路由，依然可以实现通信。这让我明白了子网掩码本质上是用于**逻辑运算以判断目标是否在本地网段**，而非物理连接的硬性屏障。

总结： 通过这次实验，我不仅熟练掌握了协议分析器、MTU 工具及命令行工具的使用，更重要的是将课本上抽象的 IP 协议格式、分片流程和路由原理转化为了可视化的数据流。这种“所见即所得”的实验方式极大地巩固了我的计算机网络知识体系。